OIPE IAP2E
APR 30 2007
PATENT & TRADEMARK OFFICE

AF
EFW

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellant:	**Proudler, Graeme John**	Examiner:	Davis, Zachary A.
Serial No.:	**09/920,554**	Art Unit:	2137
Filed:	August 01, 2001	Our Ref:	B-4240 618934-9
For:	"PERFORMANCE OF A SERVICE ON A COMPUTING PLATFORM"	Date:	April 24, 2007
		Re:	***Appeal to the Board of Appeals***

BRIEF ON APPEAL

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This is an appeal from the Final rejection, dated June 27, 2006, for the above identified patent application. The Appellants respectfully submit that the fee set forth in 37 C.F.R. 1.17(c) for submitting the present amended Appeal Brief was charged in relation with the filing of an initial Appeal Brief, on December 18, 2006. This initial Appeal Brief was eventually declared to be non-compliant in the Notice of Non-Compliant Appeal Brief issued on March 26, 2007. The Appellants submit that the present Appeal Brief is being timely filed, since it is filed in reply to the Notice of Non-Compliant Appeal Brief issued on March 26, 2007. Appellants respectfully submit that the present Appeal Brief complies with the suggestions made by telephone by the Examiner on April 19, 2007.

REAL PARTY IN INTEREST

The real party in interest to the present application is Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, California.

RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences related to the present application.

STATUS OF CLAIMS

The present Application comprises claims 1-29 and 31, which all stand rejected.

Claim 30 was cancelled without prejudice.

Claims 1-29 and 31 are the subject of this appeal and are reproduced in the accompanying appendix.

STATUS OF AMENDMENTS

There are no amendments pending in the present application.

SUMMARY OF CLAIMED SUBJECT MATTER

The invention described and claimed in the present application relates generally to the performance of services on a computing platform where reliable or trusted performance of some or all of the service is required (page 1, lines 5-7). In normal commercial life, when two parties agree that one will perform a service for another, a written contract is frequently created which specifies not only the rights and obligations of the parties but also specifies key aspects of how the service is to be performed. It is desirable also for evidence to be recorded so that it can be determined whether the service has been satisfactorily performed (page 1, lines 11-15).

The invention provides method of performing a service for a requester on a computing platform and wherein the requester provides a specification of the service to be performed to the computing platform, wherein the specification of the service establishes specified levels of trust for at least some of the processes in the service; the computing platform then executes the service according to the specification and logging performance of at least some of the processes for which a level of trust was specified; and the computing platform provides the requester with a log of the performance of the processes performed according to the specified levels of trust (page 1, lines 22-32). The invention thus allows for the

provision of evidence of satisfactory performance of services on a computing platform in response to an electronically received request (page 2, lines 1-3).

Claim 1 relates to such method, wherein the requester provides a specification of the service to be performed to the computing platform. The specification of the service establishes specified levels of trust for at least some of the processes in the service and the computing platform executes the service according to the specification and logging performance of processes for which a level of trust was specified (second and third paragraphs of claim 1). The computing platform provides the requester with a log of the performance of the processes performed according to the specified levels of trust (third paragraph of claim 1).

The Appellant wishes to emphasize that the above summary of claim 1 is by no means to be considered as a definition of the claimed invention that has any limiting effect on the scope of the claims. The scope of the claims is to be assessed solely using the language actually used in the claims.

In detail, independent claim 1 recites: "*A method of performing a service for a requestor* (801, figure 8) *on a computing platform* (401, figure 8), *comprising*:

the requestor (801, figure 8) *providing a specification* (802, figure 8) *of the service to be performed to the computing platform* (401, figure 8), *wherein the specification* (802, figure 8) *of the service establishes specified levels of trust for at least one of the processes in the service* (page 21, line 32 to page 22, line 4);

the computing platform (401, figure 8) *executing the service according to the specification and logging performance of at least one of the processes for which a level of trust was specified* (page 24, lines 19-23); *and*

the computing platform (401, figure 8) *providing the requester* (801, figure 8) *with a log of the performance of the processes performed according to the specified levels of trust* (page 28, lines 14-16 and page 29, lines 12-13)".

Preferably, the invention provides for a service management process that allocates the execution of processes and logging of performance to discrete computing environments in or associated with the computing platform (page 2, lines 22-25). These discrete computing environments may be compartments,

each containing a computing engine protected against influence from outside the compartment by operational or environmental constraints. An example of a compartment would be a Java sandbox, containing a Java Virtual Machine. Such compartments can be located inside or outside a protected computing environment, or both (page 2, lines 26-31).

Claim 24 relates to such computing platform having discrete computing environments compartments in a protected computing environment (second, third and fourth paragraphs of claim 24) protected against influence from outside the compartment and a service management process (fifth paragraph of claim 24) that allocates the execution of processes and logging of performance to the discrete computing environments.

The Appellant wishes to emphasize that the above summary of claim 24 is by no means to be considered as a definition of the claimed invention that has any limiting effect on the scope of the claims. The scope of the claims is to be assessed solely using the language actually used in the claims.

In detail, independent claim 24 recites: "*A computing platform* (10, figure 1), *comprising:*

a physically and logically protected computing environment (401, figure 8), *adapted to provide trustworthy data to appropriate users* (801, figure 8) *of the computing platform; and*

one or more compartments (805, figure 8), *arranged to operate in a sufficiently constrained manner that processes executed in a compartment are performed reliably* (page 25, lines 30-33);

wherein specified processes may be executed for a user (801, figure 8) in the one or more compartments (805, figure 8) *and the results of the specified processes returned to the user* (page 28, lines 14-16 and page 29, lines 12-13) *in trustworthy data from the protected computing environment* (401, figure 8); *and*

wherein the computing platform comprises a service management process (803, figure 8) adapted to receive a service description (802, figure 8) *which includes levels of trust assigned to processes within the service, and to allocate* (page 22, lines 8-10) *at least one of the processes to the compartments* (805, figure 8)".

* * *

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Issue 1: Whether claims 1-6, 14-26, 29 and 31 are patentable under 35 U.S.C. 103(a) over U.S. 6,289,462 to McNabb in view of U.S. 6,327,652 to England.

Issue 2: Whether claims 7-13, 27 and 28 are patentable under 35 U.S.C. 103(a) over U.S. 6,289,462 to McNabb in view of U.S. 6,327,652 to England and further in view of "HP Virtualvault Trusted Web-Server Platform Product Brief".

ARGUMENT

Issue 1: Whether claims 1-6, 14-26, 29 and 31 are patentable under 35 U.S.C. 103(a) over U.S. 6,289,462 to McNabb in view of U.S. 6,327,652 to England.

Rejection of claim 1

In the non-final Office Action mailed on February 2, 2005 the Examiner rejected claim 1 under 35 U.S.C. 102(e) under the rationale that McNabb discloses a method including a requester providing a specification of a service to be performed that establishes level of trust for processes in the service (see, for example, column 19, line 55 – column 20, line 2)". The Appellant respectfully disagrees. McNabb discloses, column 19, lines 60-62, a control method that "may be applied to a software application suite where each user is permitted to operate upon or view data at their sensitivity level", wherein the sensitivity level of the data is given by an assigned sensitivity label SL that is used "to determine if a user or process can access certain objects or resources" (column 12, lines 43-45).

In response, the Appellant disagreed with the Examiner and submitted that nowhere does McNabb suggest using the sensitivity level, which allows

determining if a process may be performed by a user, as a trust level as recited in claim 1, which indicates the "degree of reliability or security" with which a requested process must be performed (3rd line of paragraph [0072] of the Application).

Further, the Appellant noted that McNabb provides for a plurality of users or processes having a same sensitivity level to access certain objects or resources (McNabb provides at col. 18, lines 14-17 for "a set of user authorization privileges associated with the process that may describe the users individually or in a role"). In such a configuration, even a data/process with a high sensitivity level can have a low trust level for each of the users of the role allowed to access to the data/process, since any other user of the role would be able to tamper with the data/run the process. Therefore, the Appellant submitted that McNabb actually teaches away from using its "sensitivity level" as a "trust level" as recited in claim 1.

Further, the Appellant noted that McNabb discloses a method wherein the SL labels are designed to be "not under the control of the user" (Col. 9, line 16) wherein, when a requestor (user) provides a specification of a service to be performed, the specification of the service does not establish specified levels of sensitivity for the processes in the service: in Mac Nabb the levels of sensitivity for the processes in the service are assigned, for example as a function of the IP address of the user (col. 16, lines 18-19). The Appellant therefore submitted that McNabb actually teaches away from a method wherein specified levels are established under the control of a requestor, and in particular from a method wherein "the specification of the service", provided by the user, "establishes specified levels of trust for at least one of the processes in the service", as recited in claim 1.

At least in view of the above, the Appellant submitted that claim 1 is patentable over McNabb.

In a final Office Action, issued on July 25, 2005, the Examiner continued to reject claim 1 under the rationale that McNabb does in fact define the sensitivity level as "the security level of a request" (column 8, lines 33-37), which the Examiner asserts is analogous to the trust level of the present claims.

In response to this Action the Appellant responded, pursuant to a telephone conversation with the Examiner, that it is a misconception to opine that "the security level of a request" (column 8, lines 33-37), is "analogous to the trust level of the present claims". Security levels in McNabb are clearly defined as describing the sensitivity (e.g. classification) of the data of the object. (See McNabb column 8, lines 34-38 – cited by the Examiner.) McNabb fully realizes that trust is a completely different concept than is either "sensitivity level" or "security level". Note the definition which McNabb provides for a trusted computer system at column 8, lines 41-45. McNabb goes on to talk about trusted computers in his patent, but when he does so it seems to be in terms of definition given at column 8, lines 41-45. What is basically happening is that McNabb assumes away the problem of a non-trusted computer apparatus and just assumes that the system that he uses can be trusted. That is, it must have "sufficient hardware and software integrity measures that could be relied on to process sensitive or classified information..."

The present disclosure is concerned with employing integrity measures to ensure that a computer system can, in fact, be trusted.

Looking at it another way, the "security level" has to do with whether or not the user can be trusted. For example, security level asks the question "does the user have a sufficiently high security level on the computer to perform certain actions"? The level of trust works in the opposite direction. Can a user, whatever their security level might be, trust the computer that they are using to reliably "process sensitive or classified information without fear of denial of service, data theft, or corruption resulting from hostile activity" as mentioned in McNabb?

That has nothing to do with their access privilege (their security level).

The Appellant has, in the past, argued that McNabb does not disclose the claimed invention. The Examiner states otherwise, but only by asserting that the sensitivity level of requests is "analogous" to the trust level recited in the present claims (to use the Examiner's phraseology). Of course, saying one thing is analogous to another thing does not make necessarily it so and it is submitted that based on McNabb's own teaching, security and trust are completely two different concepts.

It is noted that the definition used for trust in McNabb is consistent with how the term trust is used in the present application. For example, whether or not a user can properly "trust" a computing platform can depend, for example, on many things, including whether or not the computer's BIOS has been compromised. The potential problems associated with a compromised BIOS and techniques for ensuring that the boot process is secure are discussed in the present application. See for example paragraphs [0035] through [0041] of the present application, noting that the following sentences can be found in paragraph [0041]:

"In the present embodiment, the integrity metric is acquired by the measurement function 31 by generating a digest of the BIOS instructions in the BIOS memory. Such an acquired integrity metric, if verified as described above, gives a potential user of the platform 10 a high level of confidence that the platform 10 has not been subverted at a hardware, or BIOS program, level."

The bottom line is that the Appellant basically has several issues with the Examiner's rejection of claim 1. First, by using an analogy argument, the Examiner is in essence acknowledging that this rejection under 35 U.S.C. 102 is without merit. The Examiner is in essence admitting that McNabb does not teach each and every element of the rejected claims unless some untenable analogy is made. The rejection under 35 U.S.C. 102 is improper and that rejection would at

least have to be under 35 U.S.C. 103, that is, an obviousness rejection, since by the analogy argument the Examiner is in essence changing that which McNabb teaches. However, since this rejection is really an obviousness rejection, the applicant has further issues with it, mainly:

It is not seen how it would be obvious to modify McNabb so as to handle the recited levels of trust when McNabb basically assumes that this system enjoys the highest possible level of trust (that is, there is one level of trust in McNabb).

Moreover, if McNabb can somehow be modified to meet the objection noted above, then what prior art reference(s) does the Examiner rely upon to make such a rejection? The Appellant is entitled to know the identity of the prior art and since the Examiner uses the word "analogous" in the rejection, the Appellant is assuming that the Examiner must have some knowledge of some prior art, which is not disclosed in the official action, to justify the rejection. Anyway, the Examiner is either requested (i) to cite a prior art reference supporting his contentions or, if the Examiner is relying upon "facts within the personal knowledge" of the Examiner, (ii) to provide the affidavit specified by the rules of practice (see 37 C.F.R. 1.104(d)(2)).

In the non-final Office Action issued on December 12, 2005, the Examiner rejected claim 1 for being obvious over U.S. 6,289,462 to McNabb in view of U.S. 6,327,652 to England under the rationale that "McNabb discloses a method including a requester providing a specification of a service to be performed that establishes required sensitivity level for processes in the service" (e.g. column 19, line 55 – column 20, line 2, where different processes are specified for different sensitivity levels) and "a computing platform executing the service according to the specification" (e.g. the Trusted Server of Figure 1, and Column 5, lines 20-29) and "logging performance of the processes and providing the log to the requestor" (e.g. the audit trail described at column 7, lines 28-33). The Examiner acknowledged that "McNabb does not explicitly disclose details of establishing

the trust in the computer system, nor does McNabb explicitly disclose level of trust", but opined that "England discloses a method in which an operating system is securely loaded where each component of the system is associated with a trust level (column 4, lines 5-11) and each application is also determined to be trusted or non-trusted (column 9, lines 11-20)" and further discloses "logging performances" (e.g. column 4, lines 18-23); and concluded that it would have been obvious to one of ordinary skill in the art to modify McNabb to incorporate levels of trust as taught by England, "in order to guarantee the ability to distinguish between trusted and non-trusted systems executing on the same computer (see England, column 3, lines 56-61)".

In response to this Action, the Appellant respectfully traversed the above rejection and noted that a person skilled in the art who was familiar with McNabb and with England would not jump to the conclusion made by the Examiner. England teaches otherwise. England tells the reader that digital rights management is fast becoming an essential requirement online commerce (see column 2, lines 10 – 12). England tells the reader that content providers may refuse to deliver viable online content unless there are "technologies and protocols for ensuring that digital content is properly handled in the accordance with the rights granted by the publisher." See column 2, lines 12 – 15.

Appellant noted that England then goes on to tell the reader that traditional security systems ill serve this problem. See England, column 2, line 18. England reports that there are highly secure schemes for encrypting data in networks, authenticating users, revoking certificates and storing data securely. However, England tells the reader that none of these systems address the assurance of content security after it has been delivered to a client's machine. See, for example, England, column 2, lines 10-25.

England then goes on tell the reader that there are three (3) solutions to the problem. One is to use tamper resistant boxes, a second is to use secret, propriety data formats while the third is to follow the teachings of England.

England seems to say traditional security systems, such as those taught by McNabb, do not fill the bill as far as trying to convince content providers to allow their viable digital content to be downloaded to a client's machine.

Assuming that a person skilled in the art would read the prior references cited by the Examiner, why would a person skilled in the art try to combine England and McNabb? Why go with a traditional scheme such as that offered by McNabb when England tells the reader that there is a better solution?

Accordingly, the Appellant noted that the Examiner has failed to make a Prima Facie case of obviousness. As set forth in MPEP Section 2143: "To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations."

Appellant submitted that the Examiner's rejection fails to meet these tests:

-First, there is no motivation in the prior art that the Examiner has pointed to.

-Second, there is no reasonable expectation that a person of ordinary skill could combine the references in any meaningful way. Even if a person of ordinary skill were motivated by the prior art to make the suggested combination, what, exactly, would the resulting operating system look like and would it be within the skill of a person of ordinary skill in the art to make an operating system based on such a combination? Exactly what elements disclosed in McNabb are supposed to be combined with exactly what elements in England and then why is a person skilled in the art supposed to have been motivated to make that combination?

-Third, the suggested combination fails to anticipate the claims. The

Examiner's analysis, with all due respect, pretty much ignores the language of the Claims.

Consider, for example, in Claim 1 which recites, *interalia*, "the requestor providing a specification of the service to be performed to the computing platform, wherein the specification of service establishes specified levels of trust for at least one of the processes in the service..." How is that suggested by either England or McNabb or any reasonable combination of the two? The Examiner points to column 19, line 55 through column 20, line 2 of McNabb as supposedly being pertinent to that limitation. However, in setting forth the rejection, the Examiner tries to substitute McNabb's sensitivity levels for England's trust levels. Why do that, especially considering that fact that they are not the same thing? The trust levels in England have to do with convincing a content provider to make digital content available, and not with the content provider trying to run some service on McNabb's computer! Where is there a "requestor providing a specification of the service to be performed to the computing platform" in England? And why try to marry England and McNabb? In the passage noted by the Examiner in McNabb, the sensitivity levels are concerned with the access rights of users. Note the discussion about "user accessing the system with a particular sensitivity level will be directed to a different process or data file." See column 19, lines 57 – 59. How and why is the reader motivated to change that based on England? Digital rights management is concerned with that happens to a digital file after it has been downloaded. Recall that England tells the reader that content providers may refuse to deliver viable online content unless there are "technologies and protocols for ensuring that digital content is properly handled in the accordance with the rights granted by the publisher." See column 2, lines 12 – 15 of England. England wants to make sure that "downloaded content can be protected from unauthorized access." See column 2, lines 30 – 31. McNabb is concerned with such things as controlling user access to a word processor program, for example, running on a server (note the passage noted above and cited by the Examiner). The combination of these references is clearly based on a hindsight reconstruction of appellant's claims,

opposed to a reasonably motivated combination of McNabb's and England's technologies.

Appellant noted that the Examiner's cites McNabb "in view of England in making the rejection." Thus, the Examiner seems to confer more importance on McNabb than on England. But to fair, a person of ordinary skill in the art who is presented with these two references would not know beforehand that one might be more important than the other when considering them, other than by the information presented in these documents. Since England seems to suggest not using traditional security systems, which it is submitted that McNabb falls into, it is submitted that a person of ordinary skill in the art would not try to combine these two references at all and certainly would not be motivated to come up the hindsight combination proposed by the Examiner.

Moreover, even if it were both reasonable and possible to combine these two references in some logical fashion, Appellant submitted that the combination would not anticipate Claim 1 for the reasons already discussed.

In the Final Office Action issued on June 27, 2006, the Examiner opines that the above arguments are not convincing, under the rationale that the motivation for combining England and McNabb is to be found in England since such combination would guarantee the ability to distinguish between trusted and non trusted systems executing on the same computer (See England, column 3, lines 56-61).

The Examiner further opines that "because both the McNabb and England references are directed to secure and/or trusted operated systems, and are therefore analogous art, there would be a reasonable expectation that one would be successful in combining features from the two systems".

In response to the Appellant's argument that neither England, McNabb, nor "any reasonable combination of the two" suggests the claimed limitation of "a requestor providing a specification of a service to be performed to the

computing platform, wherein the specification of service establishes specified levels of trust for at least one of the processes in the service", the Examiner notes that both McNabb and England at least suggest a requestor providing a specification of a service to be performed (e.g. column 19, line 55 to column 20, line 2 of McNabb and column 9, lines 42-51 of England, noting that a requestor provides a specification of a service, namely the downloading of specific content) and that England at least suggests levels of trust are specified for at least one process (e.g. England, column 19, lines 13-40, where trust levels specifying required functions to access certain content or processes are specified in an access control list).

The Appellant respectfully disagrees with the Examiner.

McNabb relates to a trusted computer system/server wherein the access control, rights and privileges are assigned to the individual file numbers and not strictly to the user or process that accesses the computer (column 1, lines 10-15). McNabb's trusted server system is analogous to an organization having separate divisions wherein a supervisor in one division has authority over workers in his division but may have no authority over a worker in a separate division (column 9, lines 23-33).

England relates to a subscriber computer loading a Digital Right Management Operating System (DRMOS) and issuing a certificate containing the identity of the DRMOS and data representing all the software components that are loaded and executing on the subscriber computer, wherein a content provider sever examines the certificate to determine whether it should establish a trust relationship with the DRMOS on the subscriber computer (column 9, line 60 to column 10, line 3).

The Appellant notes that even though McNabb and England relate to improving the security of an interaction between a server and a remote user/client, McNabb primarily relates to modifying the operating system of the

Server (trusted server, see claim 1 of McNabb) to make sure that a remote user cannot use any loophole of the operating system to gain unauthorized access to the server, whereas England primarily relates to modifying the operating system of the remote user/client (column 11, lines 1-4) to make sure that a desired operating system is actually loaded in the remote user/client.

The Appellant notes that client and server are well known in the art as being distinct entities having distinct operations (see for example "http://en.wikipedia.org/wiki/Client-server": characteristics of a server: passive (slave), waits for requests, upon receipt of requests, processes them and then serves replies; characteristics of a client: active (master), sends requests, waits for and receives server replies. Accordingly, the Appellant respectfully submits that it seems a bit fast to conclude that McNabb, related to modifying the operating system of a server, and England, related to modifying the operating system of a client, are "analogous art" or that "there would be a reasonable expectation that one would be successful in combining features from the two systems" just "because both the McNabb and England references are directed to secure and/or trusted operated systems".

Besides, the Examiner opines that "the motivation for combining England and McNabb is to be found in England since such combination would guarantee the ability to distinguish between trusted and non trusted systems executing on the same computer (See England, column 3, lines 56-61)". The Appellant notes that the above excerpt of England, column 3, lines 56-61, recites that "*there is a need in the art for guaranteeing that a digital rights management operating system has been properly loaded on a computer. Furthermore, such a digital rights management operating system must be readily discernable from a non-trusted operating system executing on the same computer*".

As discussed above, the "digital right management" operating system of England is loaded on a client computer.

On the other hand, McNabb relates to the operating system of a trusted

server.

The Appellant respectfully submits that the Examiner has failed to show why or how a combination of McNabb and England would "guarantee" the ability to distinguish between trusted and non trusted systems executing on the same computer. Which computer is the Examiner talking about: the server computer or the client computer?

The Appellant respectfully submits that the Examiner's rationale is unclear, and disagrees with the Examiner. However, and in order to move the Application to issue, the Appellant will now show that even if McNabb and England had been combined, they would not have led the skilled person to a method as recited in claim 1.

England discloses (column 3, lines 56-61) distinguishing between a digital rights management operating system from a non-trusted operating system executing on the same computer, wherein the computer is a client computer (column 8, lines 42-43).

McNabb discloses a secure operating system on a secure server (column 8, lines 54-58).

Accordingly, even assuming, *arguendo,* that one skilled in the art had decided to combine the teachings of McNabb and England, one would at most have obtained a system as disclosed in England having a content provider server comprising a trusted operating system as in McNabb.

The Appellant notes that in such a hypothetical system, the content provider server having the trusted operating system of McNabb would, as in England, receive from the subscriber computer/client a certificate containing the identity of the DRMOS and data representing all the software components that are loaded and executing on the subscriber computer, to allow the content

provider server to know if it should establish a trust relationship with the DRMOS on the subscriber computer.

In such a hypothetical system, the subscriber computer/client would operate as in the England reference; and the content provider server would operate as in the McNabb reference, with the exception that it would also receive/analyze a certificate to determine if it should establish a trust relationship with the subscriber computer. The Appellant notes that England teaches establishing trust or not with a subscriber and then doing business or not with the subscriber depending on the subscriber being trusted or not (having or not loaded components with a predetermined trust level). Accordingly, in a hypothetical system combining McNabb and England, if a trust relationship were established because the subscriber's certificate is acceptable, the content provider server would not operate differently than the server of McNabb to respond to service requests from the subscriber; and if no trust relationship were established, the content provider server would not execute the processes of a request from the subscriber.

The Appellant notes that if a trust relationship is established with the subscriber, there is no need to additionally associate trust levels to processes in a service request, since the subscriber (with all its loaded applications) is already trusted. Conversely, if no trust relationship is established with the subscriber, there is no need to associate trust levels to processes in a service request, since no business is done with the subscriber.

Accordingly, even if the teachings of McNabb and England were combined, the hypothetical system obtained by combining these references would still fail to disclose or suggest a method as recited in claim 1, and in particular *"providing a specification of the service to be performed to the computing platform, wherein the specification of the service establishes specified levels of trust for at least one of the processes in the service"*.

The Appellant respectfully submits that at least in view of the above, claim 1 is non-obvious over McNabb and England.

Further, the Examiner opines that McNabb discloses, "logging performance of the processes and providing the log to the requestor" (e.g. the audit trail described at column 7, lines 28-33).

The Appellant notes that column 7, lines 28-33 of McNabb recite: *"Audit trail: A set of records that collectively provide documentary evidence of processing. The audit trail enables tracing of events forward from the original transactions to related records and reports, and backward from records and reports to their component source transactions"*. The Appellant notes that McNabb does not disclose or suggest that its audit trail is a recording or logging of operation or processing performed according to a particular feature.

In particular, McNabb teaches (column 23, lines 26-35) that its trusted operating system *"employs an enhanced auditing mechanism, which allows applications to trace a wider variety of security-related events than standard operating systems. The information on these transactions is appended to an audit trail in an isolated partition, which is protected by both discretionary and mandatory access control mechanisms. This approach prevents intruders from covering their tracks and eliminating traces of penetration attempts. By protecting the audit record, sufficient evidence is maintained to support litigation and law enforcement actions"*. McNabb therefore explicitly teaches that its audit trail aims at keeping track of what operations intruders may have conducted in the trusted server. Since McNabb further teaches that unknown holes in applications or operating system software may be discovered and exploited by users with malicious intent, McNabb actually teaches recording any operation or processing, in case the operation or processing would evidence unknown holes in applications or operating system software. The Appellant notes that recording or logging indistinctly any operation/processing teaches away from only recording or logging evidence that some particular processes, which require some level of trust, have been

performed in accordance with the required level of trust; and also teaches away from *"a log of the performance of the processes performed according to the specified levels of trust"*, as recited in claim 1.

The Appellant notes that England teaches creating a boot log for each component loaded in the client computer, (column 4, lines 1-5 and 18-19) a system wherein *"the identity of an operating system running on a computer is determined from an identity associated with an initial component for the operating system, combined with identities of additional components that are loaded afterwards,"* and wherein *"a record of the loading of each component is placed into a boot log,"* whether the components have a trusted identity or an untrusted identity. Thus, England teaches logging indistinctly any processes, trusted or untrusted, and teaches away from logging the performance of processes according to the level of trust required for the processes, and consequently teaches away from *"a log of the performance of the processes performed according to the specified levels of trust"*, as recited in claim 1.

Since as detailed above both McNabb and England fail to disclose or suggest creating a log of the performance of specific processes only, and even teach away from such log or from *"a log of the performance of the processes performed according to the specified levels of trust"* as claimed in claim 1, no combination of McNabb and England would have led to a method as recited in claim 1.

It follows from the above that even assuming, *arguendo*, that one skilled in the art had found a motivation to combine McNabb and England, such combination would not have led to a method as recited in claim 1, and in particular comprising *"a log of the performance of the processes performed according to the specified levels of trust"*.

The Appellant respectfully submits that for the above reason also, claim 1 is non-obvious over McNabb and England.

The Examiner opines that both McNabb and England at least suggest a requestor providing a specification of a service to be performed (see column 19, lines 55 to column 20, line 2 of McNabb or column 9, lines 45-51 of England, where a requestor provides a specification of a service, namely the downloading of specific content). The Appellant notes that according to the analysis of the Examiner, the requestor of McNabb is a user requesting an object in the system and the requestor of England is a user requiring the downloading of specific content.

The Appellant further notes that the audit track of McNabb, deemed by the Examiner to read on the log of claim 1, is in an isolated partition protected by both discretionary and mandatory access control mechanisms to prevents intruders from covering their tracks and eliminating traces of penetration attempts to maintain sufficient evidence to support litigation and law enforcement actions. Such "log" is therefore not available to a user requesting an object in the system. Besides, the administrator that will eventually access the log will not be interested in his own log, but in the log of the other users. McNabb therefore teaches away from providing a requester with a log of the processes of its own request.

Similarly, the boot log of England is not provided to the user requesting an object, and therefore teaches away from providing a requester with a log of the processes of its own request.

It follows that for this reason also, no combination of McNabb and England would have led one skilled in the art to a method as claimed in claim 1, and in particular comprising *"providing the requester with a log of the performance of the processes performed according to the specified levels of trust"*.

The Appellant therefore submits that claim 1 is patentable over 35 U.S.C. 103, and that the Examiner's rejection should be properly overturned.

Rejection of claim 24

In the non-final Office Action mailed on February 2, 2005 the Examiner rejected claim 24 under 35 U.S.C. 102(e) under the rationale that McNabb discloses a platform including a protected computing environment (see Figure 1) and one or more compartments (column 17, lines 9-14) in which processes may be returned to the user as trustworthy data from the protected environment (see, for example, column 6, lines 20-23).

In response to this action, claim 24 was amended to recite that the computing platform comprises *"a service management process adapted to receive a service description which includes levels of trust assigned to processes within the service, and to allocate at least one of the processes to the compartments"*. The Appellant noted that the arguments used to defend claim 1, showing that contrary to the Examiner's opinion, McNabb relates to "sensitivity level" but does not suggest using such sensitivity level as a trust level, can be used to show that McNabb does neither disclose or suggest, and actually teaches away from a computing platform as recited in claim 24, and in particular comprising *" a service management process adapted to receive a service description which includes levels of trust assigned to processes within the service, and to allocate at least one of the processes to the compartments"*. The Appellant submitted that at least in view of the above, claim 24 is patentable over McNabb.

In a final Office Action, issued on July 25, 2005, the Examiner kept rejecting claim 24 under the rationale that McNabb does in fact define the sensitivity level as "the security level of a request" (column 8, lines 33-37), which is therefore analogous to the trust level of the present claims.

In response to this Action the Appellant responded, pursuant to a telephone conversation with the Examiner, that it is a misconception to opine that "the security level of a request" (column 8, lines 33-37), is "analogous to the

trust level of the present claims". Security levels in McNabb are clearly defined as describing the sensitivity (e.g. classification) of the data of the object. (see McNabb column 8, lines 34-38 – cited by the Examiner). McNabb fully realizes that trust is a completely different concept than is either "sensitivity level" or "security level". Note the definition which McNabb provides for a trusted computer system at column 8, lines 41-45. McNabb goes on to talk about trusted computers in his patent, but when he does so it seems to be in terms of definition given at column 8, lines 41-45. What is basically happening is that McNabb assumes away the problem of a non-trusted computer apparatus and just assumes that the system which he uses can be trusted. That is, it must have "sufficient hardware and software integrity measures that could be relied on to process sensitive or classified information..."

The present disclosure is concerned with employing integrity measures to ensure that a computer system can, in fact, be trusted.

Looking at it another way, the "security level" has to do with whether or not the user can be trusted. For example, security level asks the question "does the user have a sufficiently high security level on the computer to perform certain actions"? The level of trust works in the opposite direction. Can a user, whatever their security level might be, trust the computer that they are using to reliably "process sensitive or classified information without fear of denial of service, data theft, or corruption resulting from hostile activity" as mentioned in McNabb ? That has nothing to do with their access privilege (their security level).

The Appellant has, in the past, argued that McNabb does not disclose the claimed invention. The Examiner states otherwise, but only by asserting that the sensitivity level of requests is "analogous" to the trust level recited in the present claims (to use the Examiner's phraseology). Of course, saying one thing is analogous to another thing does not make necessarily it so and it is submitted that based on McNabb's own teaching, security and trust are completely two different concepts.

It is noted that the definition used for trust in McNabb is consistent with how the term trust is used in the present application. For example, whether or not a user can properly "trust" a computing platform can depend, for example, on many things, including whether or not the computer's BIOS has been compromised. The potential problems associated with a compromised BIOS and techniques for ensuring that the boot process is secure are discussed in the present application. See for example paragraphs [0035] through [0041] of the present application, noting that the following sentences can be found in paragraph [0041]:

"In the present embodiment, the integrity metric is acquired by the measurement function 31 by generating a digest of the BIOS instructions in the BIOS memory. Such an acquired integrity metric, if verified as described above, gives a potential user of the platform 10 a high level of confidence that the platform 10 has not been subverted at a hardware, or BIOS program, level."

The Appellant further noted that McNabb does not teach "service management process adapted to receive a service description which includes levels of trust assigned to process this within the service ..." and that as indicated above, McNabb knows nothing about levels of trust other than to assume that the computer platform in question has "sufficient hardware and software integrity measures" without reference to "levels of trust".

In the non-final Office Action issued on December 12, 2005, the Examiner rejected claim 24 under 35 U.S.C. 103(a) as being unpatentable over McNabb in view of England, under the rationale that McNabb discloses a platform including a protected computing environment (see Figure 1) and one or more compartments (column 17, lines 9-14), in which processes may be returned to the user as trustworthy data from the protected environment (see, for example, column 6, lines 20-23), and where the platform further includes a management process that receives a service description including required sensitivity levels for processes within the service (see, for example, column 19, line 55 – column 20,

line 2, where different processes are specified for different sensitivity levels) and that allocates the processes to the compartments (column 21, lines 34-55). However, although McNabb discloses sensitivity levels that describe required security (column 8, lines 33-37 and 10-15) and that there is a trusted computer system (column 8, lines 40-45), McNabb does not explicitly disclose details of establishing the trust in the computer system, nor does McNabb explicitly disclose levels of trust.

England discloses a system in which an operating system is securely loaded where each component of the system is associated with a trust level (column 4, lines 5-11) and each application is also determined to be trusted or non-trusted (column 9, lines 11-20). Therefore, it would have been obvious to one of ordinary skill in the art to modify the platform of McNabb to incorporate levels of trust as taught by England, in order to guarantee the ability to distinguish between trusted and non-trusted systems executing on the same computer (see England, column 3, lines 56-61).

In response to this action, the Appellant argued that a person skilled in the art who was familiar with McNabb and with England would not jump to the conclusion made the Examiner, for example because there was no motivation for combining McNabb and England, as detailed above in relation to claim 1.

In the Final Office Action issued on June 27, 2006, the Examiner opines that the above arguments are not convincing, under the rationale that the motivation for combining England and McNabb is to be found in England since such combination would guarantee the ability to distinguish between trusted and non trusted systems executing on the same computer (See England, column 3, lines 56-61). As seen above in relation with claim 1, the Examiner further opines that "because both the McNabb and England references are directed to secure and/or trusted operated systems, and are therefore analogous art, there would be a reasonable expectation that one would be successful in combining features from the two systems".

The Appellant respectfully disagrees with the Examiner. As detailed above in relation with claim 1, McNabb relates to a trusted computer system/server wherein the access control, rights and privileges are assigned to the individual file numbers and not strictly to the user or process that accesses the computer (column 1, lines 10-15), and substantially teaches modifying the operating system of the server to make the server a trusted server. On another hand, England relates to a subscriber computer loading a Digital Right Management Operating System (DRMOS) and issuing a certificate containing the identity of the DRMOS and data representing all the software components that are loaded and executing on the subscriber computer to allow a content provider server to know if it should establish a trust relationship with the DRMOS on the subscriber computer (column 9, line 60 to column 10, line 3).

Accordingly, the Appellant respectfully submit that it seems doubtful that McNabb, related to modifying the operating system of a server, and England, related to modifying the operating system of a client, are "analogous art" or that "there would be a reasonable expectation that one would be successful in combining features from the two systems", even though both the McNabb and England references are directed to secure and/or trusted operated systems.

The Examiner opines that, "the motivation for combining England and McNabb is to be found in England since such combination would guarantee the ability to distinguish between trusted and non trusted systems executing on the same computer." As detailed above, the Appellant respectfully disagrees with the Examiner. However, and in order to move the Application to issue, the Appellant will now show that even if McNabb and England had been combined, they would not have led the skilled person to a platform as recited in claim 24.

England discloses (column 3, lines 56-61) distinguishing between a digital rights management operating system from a non-trusted operating system executing on the same computer, wherein the computer is a client computer

(column 8, lines 42-43).

McNabb discloses a secure operating system on a secure server (column 8, lines 54-58).

Accordingly, even assuming, *arguendo,* that one skilled in the art had decided to combine the teachings of McNabb and England, one would at most have obtained a system as disclosed in England having a content provider server comprising a trusted operating system as in McNabb.

The Appellant notes that in such a hypothetical system, the content provider server having the trusted operating system of McNabb would, as in England, receive from the subscriber computer/client a certificate containing the identity of the DRMOS and data representing all the software components that are loaded and executing on the subscriber computer to allow the content provider server to know if it should establish a trust relationship with the DRMOS on the subscriber computer.

In such a hypothetical system, the subscriber computer/client would operate as in the England reference; and the content provider server would operate as in the McNabb reference, with the exception that it would also receive /analyze a certificate to determine if it should establish a trust relationship with the subscriber computer. The Appellant notes that England teaches establishing trust or not with a subscriber and then doing business or not with the subscriber depending on the subscriber being trusted or not (having or not loaded components with a predetermined trust level). Accordingly, in a hypothetical system combining McNabb and England, if a trust relationship were established because the subscriber's certificate is acceptable, the content provider server would not operate differently than the server of McNabb to respond to service requests from the subscriber; and if no trust relationship were established, the content provider server would not execute the processes of a request from the subscriber.

The Appellant notes that if a trust relationship is established with the subscriber, there is absolutely no need to additionally associate trust levels to processes in a service request, since the subscriber (with all its loaded applications) is already trusted. Conversely, if no trust relationship is established with the subscriber, there is absolutely no need to associate trust levels to processes in a service request, since no business is done with the subscriber.

Accordingly, even if the teachings of McNabb and England were combined, the hypothetical system obtained by combining these references would still fail to disclose or suggest a computing platform as recited in claim 24, and in particular comprising, *"a service management process adapted to receive a service description which includes levels of trust assigned to processes within the service, and to allocate at least one of the processes to the compartments"*.

The Appellant respectfully submits that at least in view of the above, claim 24 is non-obvious over McNabb and England. Appellant therefore respectfully submits that claim 24 is patentable over 35 U.S.C. 103, and that the Examiner's rejection should be properly overturned.

Rejection of claims 2-6, 14-23, 25-26, 29 and 31

Claims 2-6 and 14-26 depend directly or indirectly on claim 1, and claims 25-26 and 29 depend on claim 24. The Appellant respectfully submits that at least in view of their dependency, claims 2-6, 14-23, 25-26, 29 and 31 are patentable over McNabb and England and thus comply with 35 U.S.C. 103, whereby the Examiner's rejection should be properly overturned.

Issue 2: Whether claims 7-13, 27 and 28 are patentable under 35 U.S.C. 103(a) over U.S. 6,289,462 to McNabb in view of U.S. 6,327,652 to England and further in view of "HP Virtualvault Trusted Web-Server Platform Product Brief".

Claims 7-13 depend on claim 1, and claims 27-28 depend on claim 24. The Appellant submits that the Examiner has failed to show that Virtualvault discloses or suggests a method as recited in claim 1, and in particular comprising: "*the requestor providing a specification of the service to be performed to the computing platform, wherein the specification of the service establishes specified levels of trust for at least one of the processes in the service*", or a computing platform as recited in claim 24, and in particular, comprising: "*a service management process adapted to receive a service description which includes levels of trust assigned to processes within the service, and to allocate at least one of the processes to the compartments*". Accordingly, the Appellant submits that no combination of McNabb and Virtualvault would have led one skilled in the art to a method as recited in claim 1 or to a computing platform as recited in claim 24, and that both claims 1 and 24 are patentable over McNabb in view of VirtualVault. The Appellant further submits that at least in view of their dependency on claims 1 or 24, claims 7-13 and claims 27-28 are patentable over McNabb in view of Virtualvault.

* * *

CONCLUSION

For the extensive reasons advanced above, Appellants respectfully contend that each claim is patentable. Therefore, reversal of the above-addressed rejections and objections and re-opening of the prosecution is respectfully solicited.

The Commissioner is authorized to charge any additional fees that may be required or credit overpayment to deposit account no. 08-2025. In particular, if this response is not timely filed, the Commissioner is authorized to treat this response as including a petition to extend the time period pursuant to 37 CFR 1.136(a) requesting an extension of time of the number of months necessary to make this response timely filed and the petition fee due in connection therewith may be charged to deposit account no. 08-2025.

I hereby certify that this correspondence is being deposited with the United States Post Office with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on

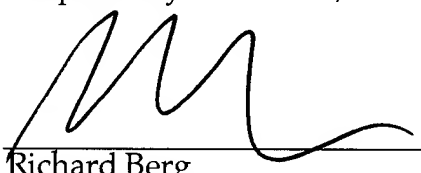
April 24, 2007
(Date of Transmission)

Diane Osollo
(Name of Person Transmitting)

Diane Osollo
(Signature)

April 24, 2007
(Date)

Respectfully submitted,



Richard Berg
Attorney for the Appellant
Reg. No. 28,145
LADAS & PARRY
5670 Wilshire Boulevard,
Suite 2100
Los Angeles, California 90036
(323) 934-2300 voice
(323) 934-0202 facsimile

Attachments: Claims 1-29 and 31

CLAIMS APPENDIX

1. A method of performing a service for a requestor on a computing platform, comprising:

the requestor providing a specification of the service to be performed to the computing platform, wherein the specification of the service establishes specified levels of trust for at least one of the processes in the service;

the computing platform executing the service according to the specification and logging performance of at least one of the processes for which a level of trust was specified; and

the computing platform providing the requester with a log of the performance of the processes performed according to the specified levels of trust.

2. A method as claimed in claim 1, wherein a level of trust is specified for at least two processes in the specification, and no performance logging takes place for at least one of the processes for which a level of trust is specified in the specification.

3. A method as claimed in claim 1, wherein the computing platform contains a physically and logically protected computing environment.

4. A method as claimed in claim 3, wherein said physically and logically protected computing environment contains a monitoring process for measuring integrity of the computing platform.

5. A method as claimed in claim 3, wherein a service management process allocates the execution of processes and logging of performance to discrete computing environments in or associated with the computing platform.

6. A method as claimed in claim 5, wherein the service management

process is located within the protected computing environment.

7. A method as claimed in claim 5, wherein one or more of the discrete computing environments is a compartment containing a computing engine protected against influence from outside the compartment by operational or environmental constraints.

8. A method as claimed in claim 7, wherein the computing engine is a Java virtual machine.

9. A method as claimed in claim 7, wherein one or more compartments is located within the protected computing environment.

10. A method as claimed in claim 7, wherein the computing engine is constrained not to operate on input data if it is not permitted to do so.

11. A method as claimed in claim 10, wherein input data is provided with a data type, and a process is provided with operation types, and operation is prevented if operation types and data types are not consistent.

12. A method as claimed in claim 10, wherein input data may have an owner, and the process may be required to inform the owner of use of the input data.

13. A method as claimed in claim 10, wherein input data may have an owner, and if so, the process may be required to obtain consent from the owner to use the input data.

14. A method as claimed in claim 5, wherein a process may be swapped between one discrete environment and another discrete environment.

15. A method as claimed in claim 1, wherein performance logging includes

logging of input data to a process.

16. A method as claimed in claim 1, wherein performance logging includes logging of output data from a process.

17. A method as claimed in claim 1, wherein performance logging includes logging of program instructions executed in performance of a process.

18. A method as claimed in claim 1, wherein data logged is sampled according to a sampling process to provide the performance log.

19. A method as claimed in claim 18, wherein the sampling process is performed according to a function to provide irregular sampling.

20. A method as claimed in claim 1, where a digest of data logged is obtained as part of the performance logging data.

21. A method as claimed in claim 1, wherein the performance logging data is encrypted before it is sent to the requestor.

22. A method as claimed in claim 1, wherein the specification establishes performance logging parameters for at least one of the processes in the service.

23. A method as claimed in claim 4, wherein the monitoring process provides to the requestor an integrity metric of the computing platform at the time the service was performed.

24. A computing platform, comprising:

a physically and logically protected computing environment, adapted to provide trustworthy data to appropriate users of the computing platform; and

one or more compartments, arranged to operate in a sufficiently constrained manner that processes executed in a compartment are performed

reliably;

wherein specified processes may be executed for a user in the one or more compartments and the results of the specified processes returned to the user in trustworthy data from the protected computing environment; and

wherein the computing platform comprises a service management process adapted to receive a service description which includes levels of trust assigned to processes within the service, and to allocate at least one of the processes to the compartments.

25. A computing platform as claimed in claim 24, wherein one or more of said compartments are located outside the protected computing environment.

26. A computing platform as claimed in claim 24, wherein one or more of said compartments are located inside the protected computing environment.

27. A computing platform as claimed in claim 24, wherein each compartment contains a virtual computing engine.

28. A computing platform as claimed in claim 27, wherein the virtual computing engine is a Java virtual machine.

29. A computing platform as claimed in claim 24, wherein the protected computing environment contains a monitoring process adapted to measure the integrity of the computing platform.

30. (cancelled)

31. A computing platform as claimed in claim 24, wherein service management process is located within the protected computing environment.

EVIDENCE APPENDIX

There is no evidence submitted with the present Appeal Brief.

RELATED PROCEEDINGS APPENDIX

There are no other appeals or interferences related to the present application.